CLAIMS

1   What is claimed is

1.   A system for determining performance of an application under test in response to
2   load, the system comprising:
3   a)   coordination software;
4   b)   at least one code generator, receiving as an input commands from the
5   coordination software and having as an output client test code;
6   c)   at least one test engine, receiving as an input commands from the
7   coordination software, the test engine comprising a computer server having at least one
8   JVM executing at least one instance of the client test code;
9   d)   at lease one data log having computerized memory, the memory holding
10   timing data created by the instances of the client test code ; and
11   e)   at least one data analyzer software, operatively connected to the data log,
12   having an output that represents performance of the application under test in response to
13   load.

1   2.   The system of claim 1 wherein said at least one JVM executes multiple threads,
2   each thread comprising an instance of the client test code.

1   3.   The system of claim 2 wherein said at least one JVM is synchronized to start
2   execution of an instance of the client test code with another of said at least one JVM
3   about to start execution an instance of the client test code.

1   4.   The system of claim 3 wherein the synchronization of at least one JVM to another
2   of said at least one JVM is performed independently of the time set on each system.

1   5.   The system of claim 3 wherein said at least one JVM is set to start execution of
2   the client test code a predetermined time after another of said at least one JVM is set to
3   start execution of the test client code.

1   6.   The system of claim 2 wherein said at least one JVM is set to start execution of
2   the client test code independent of another of said at least one JVM set to start execution
3   of the client test code.

1   7.   A computer program product for determining performance of an application under

2 test in response to load, the computer program product comprising a computer usable

3 medium having computer readable code thereon, including program code comprising:

4   a)   instructions for coordination software;

5   b)   instructions for at least one code generator, receiving as an input

6 commands from the coordination software and having as an output client test code;

7   c)   instructions for at least one test engine, receiving as an input commands

8 from the coordination software, the test engine comprising a computer server having at

9 least one JVM executing at least one instance of the client test code;

10   d)   instructions for providing at lease one data log having computerized

11 memory, the memory holding timing data created by the instances of the client test code;

12 and

13   e)   instructions for providing at least one data analyzer, operatively connected

14 to the data log, having an output that represents performance of the application under test

15 in response to load.

1 8.   The computer program product of claim 7 further comprising instructions for

2 causing said at least one JVM to execute multiple threads, each thread comprising an

3 instance of the client test code.

1 9.   The computer program product of claim 7 further comprising instructions for

2 causing said at least one JVM to be synchronized to start execution of an instance of the

3 client test code with another of said at least one JVM about to start execution an instance

4 of the client test code.

1 10.   The computer program product of claim 9 further comprising instructions wherein

2 the synchronization of at least one JVM to another of said at least one JVM is performed

3 independently of the time set on each system.

1 11.   The computer program product of claim 9 further comprising instructions wherein

2 said at least one JVM is set to start execution of the client test code a predetermined time

3 after another of said at least one JVM is set to start execution of the test client code.

1 12.   The computer program product of claim 8 further comprising instructions wherein

2 said at least one JVM is set to start execution of the client test code independent of

3 another of said at least one JVM set to start execution of the client test code.

1 13. A method of testing a computerized application, the application under test having
2 a plurality of software components, at least one component having at least one method
3 therein, the method comprising the steps of:

    4   a)   providing test code to exercise a component;

    5   b)   creating a plurality of copies of the test code;

    6   c)   simultaneously executing the plurality of copies of the test code;

    7   d)   providing a folder for each method of the component being exercised;

    8   e)   recording times for each method of the component being exercised; and

    9   f)   analyzing the recorded times to present information on performance of
10 each method of the component being exercised.

1 14. The method of claim 13 wherein the components are selected from Enterprise
2 Java Beans and Component Object Modules.

1 15. The method of claim 13 wherein said step of recording times further comprises
2 recording said times for each method in a respective folder for said method.

1 16. The method of claim 15 wherein each folder is used to provide calculations for
2 said method from said times recorded in the folder.

1

1 17. The method of claim 16 wherein said calculations are selected from the group
2 consisting of the average response time of the items within the folder, and the total
3 response time of the items within the folder.

1 18. A computer program product for testing a computerized application, the
2 application under test having a plurality of software components, at least one component
3 having at least one method therein, the computer program product comprising a computer
4 usable medium having computer readable code thereon, including program code
5 comprising:

    6   a)   instructions for providing test code to exercise a component;

    7   b)   instructions creating a plurality of copies of the test code;

    8   c)   instructions for simultaneously executing the plurality of copies of the test
9 code;

10   d)   instructions for providing a folder for each method of the component
11 being exercised;

12   e)   instructions for recording times for each method of the component being
13   exercised; and
14   f)   instructions for analyzing the recorded times to present information on
15   performance of each method of the component being exercised.

1   19.   The computer program product of claim 18 wherein the components are selected
2   from Enterprise Java Beans and Component Object Modules.

1   20.   The computer program product of claim 18 further comprising instructions for
2   recording said times for each method in a respective folder for said method.

1   21.   The computer program product of claim 20 further comprising instructions for
2   providing calculations for said method from said times recorded in the folder.

1   22.   The computer program product of claim 21 for causing said calculations to be
2   selected from the group consisting of the average response time of the items within the
3   folder, and the total response time of the items within the folder.

1   23.   A system for determining performance of an application under test in response to
2   load, the system comprising:
3   a)   coordination software;
4   b)   at least one code generator, receiving as an input commands from the
5   coordination software and having as an output client test code, said code generator
6   providing a template for a datatable, said datatable used to provide information for
7   exercising the application under test;
8   c)   at least one test engine, receiving as an input commands from the
9   coordination software, the test engine comprising a computer server having a plurality of
10   threads thereon, each thread executing an instance of the client test code;
11   f)   at lease one data log having computerized memory, the memory holding
12   timing data created by the instances of the client test code in the plurality
13   of threads; and
14   g)   at least one data analyzer software, operatively connected to the data log,
15   having an output that represents performance of the application under test
16   in response to load.

1   24.   The system of claim 23 wherein said datatable includes a plurality of rows and a

2 plurality of columns wherein said columns are used for parameters and said rows

3 represent users.

1 25. The system of claim 23 wherein said datatable is in a .CSV format.

1 26. The system of claim 23 wherein when said datatable contains fewer rows than the

2 number of virtual users provided by said test engine, then said test code will cycle

3 through said data table and then start over beginning with the first row of said datatable.

1 27. A computer program product for determining performance of an application under

2 test in response to load, the computer program product comprising a computer usable

3 medium having computer readable code thereon, including program code comprising: ·

4   a) instructions for coordination software;

5   b) instructions for at least one code generator, receiving as an input

6 commands from the coordination software and having as an output client test code, said

7 code generator providing a template for a datatable, said datatable used to provide

8 information for exercising the application under test;

9   c) instructions for at least one test engine, receiving as an input commands

10 from the coordination software, the test engine comprising a computer server having a

11 plurality of threads thereon, each thread executing an instance of the client test code;

12   d) instructions for providing at lease one data log having computerized

13 memory, the memory holding timing data created by the instances of the client test code

14 in the plurality of threads; and

15   e) instructions for providing at least one data analyzer software, operatively

16 connected to the data log, having an output that represents performance of the application

17 under test in response to load.

1 28. The computer program product of claim 27 wherein said datatable includes a

2 plurality of rows and a plurality of columns wherein said columns are used for parameters

3 and said rows represent users.

1 29. The computer program product of claim 27 wherein said datatable is in a .CSV

2 format.

1 30. The computer program product of claim 28 wherein when said datatable contains

2 fewer rows than the number of virtual users provided by said test engine, then said test

3 code will cycle through said data table and then start over beginning with the first row of

4 said datatable.

1 31. A method of testing a computerized application under test that allows

2 simultaneous users over a computer network, the application under test having a plurality

3 of software components, the method comprising the steps of:

4   a)   providing test code to exercise a component, said component including at

5 least one method;

6   b)   providing a class file for each method of said component directly to each

7 user;

8   c)   creating a first plurality of copies of the test code;

9   d)   simultaneously executing the first plurality of copies of test code while

10 recording times between events in each of the first plurality of copies of test code;

11   e)   creating a second plurality of copies of test code,

12   f)   simultaneously executing the second plurality of copies of test code while

13 recording times between events in each of the second plurality of copies of test code;

14   g)   repeating a predetermined number of times the steps of creating plural

15 copies of the test code and simultaneously executing the plural copies while recording

16 event times; and

17   h)   analyzing the recorded times to present information on the performance of

18 the component of the application under test as a function of load.

1 32. The method of claim 31 wherein said class file is provided as a compressed file.

1 33. The method of claim 32 wherein said compressed file comprises a Java Archive

2 file.

1 34. A computer program product for testing a computerized application under test that

2 allows simultaneous users over a computer network, the application under test having a

3 plurality of software components, the computer program product comprising a computer

4 usable medium having computer readable code thereon, including program code

5 comprising:

6   instructions for providing test code to exercise a component, said component

7 including at least one method;

8   instructions for providing a class file for each method of said component directly

9 to each user;

10   instructions for creating a first plurality of copies of the test code;

11   instructions for simultaneously executing the first plurality of copies of test code

12   while recording times between events in each of the first plurality of copies of test code;

13   instructions for creating a second plurality of copies of test code;

14   instructions for simultaneously executing the second plurality of copies of test

15   code while recording times between events in each of the second plurality of copies of

16   test code;

17   instructions for repeating a predetermined number of times the steps of creating

18   plural copies of the test code and simultaneously executing the plural copies while

19   recording event times; and

20   instructions for analyzing the recorded times to present information on the

21   performance of the component of the application under test as a function of load.

1   35.   The computer program product of claim 34 including instructions for causing said

2   class file to be provided as a compressed file.

1   36.   The computer program product of claim 34 including instructions for causing said

2   class file to be provided as a Java Archive file.